CLAIMS

1. A scheduling method for a multi-level class hierarchy wherein classes are represented as nodes, said method comprising:

selecting non-priority nodes of said unified tree to establish a non-priority sub-tree;

selecting priority nodes of said unified tree to establish one or more priority sub-trees corresponding to one or more priority levels;

if and only if queues of nodes of said priority sub-trees are empty, applying a first scheduling algorithm to said non-priority sub-tree to select a packet for transmission; and

if any of said one or more priority sub-trees are non-empty, selecting a priority packet from said one or more priority sub-trees for transmission.

2. The scheduling method of claim 1 wherein selecting a packet from said one or more priority sub-trees for transmission comprises:

selecting a highest priority non-empty sub-tree from said one or more priority sub-trees; and

applying a second scheduling algorithm to said highest priority non-empty sub-tree to select a priority packet for transmission.

3. The scheduling method of claim 2 further comprising:

updating scheduling state within said non-priority sub-tree to reflect transmission of said priority packet.

4. The scheduling method of claim 3 wherein updating comprises:

identifying a node within said non-priority sub-tree that has a parent relationship, as viewed in said multi-level class hierarchy, to a node within said selected priority sub-tree associated with said transmitted priority packet; and

updating a scheduling state of said identified node and ancestor nodes of said identified node within said non-priority sub-tree.

5. The scheduling method of claim 4 wherein identifying comprises:

employing a pointer to said identified node.

6. The scheduling method of claim 4 wherein updating said scheduling state comprises:

adding a length of said selected priority packet to a length of a next transmitted packet associated with said identified node to be used in making further scheduling decisions within said non-priority sub-tree.

7. A computer program product for scheduling a multi-level class hierarchy wherein classes are represented as nodes, said product comprising:

code that causes selection of non-priority nodes of said unified tree to establish a non-priority sub-tree;

code that causes selection of priority nodes of said unified tree to establish one or more priority sub-trees corresponding to one or more priority levels;

code that, if and only if queues of nodes of said priority sub-trees are empty, causes application of a first scheduling algorithm to said non-priority sub-tree to select a packet for transmission;

code that, if any of said one or more priority sub-trees are non-empty, causes selection of a priority packet from said one or more priority sub-trees for transmission; and

a computer-readable medium that stores the codes.

8. The product of claim 7 wherein said code that causes selection of a packet from said one or more priority sub-trees for transmission comprises:

code that causes selection of a highest priority non-empty sub-tree from said one or more priority sub-trees; and

code that causes application of a second scheduling algorithm to said highest priority non-empty sub-tree to select a priority packet for transmission.

9. The product of claim 8 further comprising:

code that causes updating of scheduling state within said non-priority sub-tree to reflect transmission of said priority packet.

10. The product of claim 9 wherein said code that causes updating comprises:

code that causes identification of a node within said non-priority sub-tree that has a parent relationship, as viewed in said multi-level class hierarchy, to a node within said selected priority sub-tree associated with said transmitted priority packet; and

code that causes updating of scheduling state of said identified node and ancestor nodes of said identified node within said non-priority sub-tree.

11. The product of claim 10 wherein said code that causes identification comprises:

code that causes employment of a pointer to said identified node.

12. The product of claim 10 wherein said code that causes updating of said scheduling state comprises:

code that causes addition of a length of said selected priority packet to a length of a next transmitted packet associated with said identified node to be used in making further scheduling decisions within said non-priority sub-tree.

13. Apparatus for scheduling a multi-level class hierarchy wherein classes are represented as nodes, said apparatus comprising:

a processor; and

a memory storing instructions for execution by said processor, said instructions comprising;

code that causes selection of non-priority nodes of said unified tree to establish a non-priority sub-tree;

code that causes selection of priority nodes of said unified tree to establish one or more priority sub-trees corresponding to one or more priority levels;

code that, if and only if queues of nodes of said priority sub-trees are empty, causes application of a first scheduling algorithm to said non-priority sub-tree to select a packet for transmission; and

code that, if any of said one or more priority sub-trees are non-empty, causes selection of a priority packet from said one or more priority sub-trees for transmission.

14. The apparatus of claim 13 wherein said code that causes selection of a packet from said one or more priority sub-trees for transmission comprises:

code that causes selection of a highest priority non-empty sub-tree from said one or more priority sub-trees; and

code that causes application of a second scheduling algorithm to said highest priority non-empty sub-tree to select a priority packet for transmission.

15. The apparatus of claim 14 wherein said instructions further comprise:

code that causes updating of scheduling state within said non-priority sub-tree to reflect transmission of said priority packet.

16. The apparatus of claim 15 wherein said code that causes updating comprises:

code that causes identification of a node within said non-priority sub-tree that has a parent relationship, as viewed in said multi-level class hierarchy, to a node within said selected priority sub-tree associated with said transmitted priority packet; and

code that causes updating of scheduling state of said identified node and ancestor nodes of said identified node within said non-priority sub-tree.

17. The product of claim 16 wherein said code that causes identification comprises:

code that causes employment of a pointer to said identified node.

18. The apparatus of claim 16 wherein said code that causes updating of said scheduling state comprises:

code that causes addition of a length of said selected priority packet to a length of a next transmitted packet associated with said identified node to be used in making further scheduling decisions within said non-priority sub-tree.

19. Apparatus for scheduling a multi-level class hierarchy wherein classes are represented as nodes, said apparatus comprising:

means for selecting non-priority nodes of said unified tree to establish a non-priority sub-tree;

means for selecting priority nodes of said unified tree to establish one or more priority sub-trees corresponding to one or more priority levels;

means for, if and only if queues of nodes of said priority sub-trees are empty, applying a first scheduling algorithm to said non-priority sub-tree to select a packet for transmission; and

means for, if any of said one or more priority sub-trees are non-empty, selecting a priority packet from said one or more priority sub-trees for transmission.